Software Requirements and Specifications

for

EWU Online Course Evaluator

Version 1.3

Prepared by Sumayah Almohanna, Regan Fencl, Mick Warren, Lenna Wenke

Eastern Washington University Computer Science Department

Senior Project, Winter 2014

1. Introduction

1.1 Purpose

This SRS document lays out the functional and non-functional requirements for the EWU Online Course Evaluation Web Application. This document is to be a reference for the client (EWU Computer Science Department) and development teams.

1.2 Project Scope and Product Features

The Computer Science Department provides an opportunity for students to evaluate faculty each quarter for each course. The project is to create an online application that would allow students to evaluate their courses during specified evaluation time periods each quarter. In addition, staff identified as admins will have access to features such as determining admins, uploading course information, creating, modifying, or adding questions, and viewing evaluations, as well as generating reports and deleting data regarding instructors that is no longer needed. Instructors will be able to define, modify, and delete course-specific questions and determine which course-specific and optional questions will also be a part of a course evaluation.

2. Overall Description

2.1 Product Perspective

This application is to provide an online solution for physical course evaluations required at the university. It will reduce paper waste, greatly reduce clerical time compiling the data, and protect student anonymity for their evaluations, as there is no handwriting for anyone to recognize.

2.2 User Classes and Characteristics

Student – This user is a person is who is currently taking one or more Computer Science courses which can be evaluated during the specified evaluation periods.

Admin – This user is in an administrative position whose goal is to generate reports for the courses during or after their evaluation periods. In addition, an Admin can manage other Admin users, upload course information, create, modify, and delete questions, add questions for the instructors, view evaluations, and delete data pertaining to a particular instructor.

Instructor – This user is a person who is currently teaching Computer Science courses. This user has the ability to determine the optional and course-specific questions to be included their course evaluations for the quarter, as well as manage course-specific questions by adding, modifying, or deleting questions.

2.3 Operating Environment

OE-1: The web application shall be accessible from all the major browsers: Internet Explorer (version 6 or higher), Safari (version 5.1 or higher), the current version of Firefox, and the current version of Google Chrome.

OE-2: The web application will be hosted on a LAMP server, which has Linux, Apache, MySQL, and PHP installed.

OE-3: Students, Admins, and Instructors shall have access to the web application on the EWU campus network. To access the web application off campus, Admins and Instructors must be connected to *https://vpn.ewu.edu* . Students must be connected to *https://studenvpn.ewu.edu* .

2.4 Design and Implementation Constraints

CO-1: The web application must be hosted on a LAMP server to avoid licensing fees.

CO-2: The web application must use the SSO (Single Sign On) login page to authenticate a user, so the PHP CAS (Central Authentication Service) library must be used in order to initiate SSO.

CO-3: This web application must use the ODATA service provided by the Office of Information Technology at Eastern. Thus the “curl” and “xsl” libraries must be installed and the PHP ODATA SDK must be used in order to contact the service to query for the necessary information.

CO-3: The web application is a three-tiered application whose development is structured using the CodeIgniter framework.

2.5 Assumptions and Dependencies

DE-1: The validation of users will depend upon the functioning of SSO.

DE-2: The information provided by the ODATA Service is maintained by the Office of Information Technology, and it is this information that determines what subjects that web application will be checking when the website is in use.

2.6 Technologies

T-1: CodeIgniter Framework

This framework for PHP development was chosen for its adherence to the MVC pattern of three-tiered development and the support it provides to make MVC development easier. In addition, CodeIgniter provides the support necessary for securing session data, preventing direct script access, and scrubbing queries for SQL injection, as well as a better way to redirect between web pages.

T-2: JQuery Library

This JavaScript library was chosen due to the functionality it provides in regard to user interactions and dynamic refreshing of the client-side browser.

T-3: MPDF

This PHP PDF library was chosen for the ease with which PDFs can be generated by using defined HTML elements.

T-4: ODATA

The PHP SDK of this library was necessary for retrieving course and instructor information from the ODATA service maintained by the Office of Information Technology.

T-5: CAS

The PHP CAS (Central Authentication Service) library was required, because our web application needs to contact Eastern's CAS server in order to authenticate users through the SSO (Single Sign On) login page.

3. System Specific Descriptions

3.1 Login

A user who wishes to log in must be authenticated through the SSO login page, and thus the user will be redirected to that page when they attempt to reach the base url of: cs-course-evals.ewu.edu/online_eval. Upon being authenticated by SSO and checked by the web application to see if the user has the role of student, instructor, or admin, the user will directed to their appropriate home page based on their role(s). There is a priority system in place to direct the user to a specific page if the user has multiple roles. An example would be a Graduate Student, who can be both an instructor and a student. So by the priority ranking "Admin > Instructor > Student", a Graduate Student would be redirected to the home page of an Instructor.

3.2 Student Home

A Student user, who has logged in through SSO and has Computer Science courses to evaluate, will be presented with the courses they are to evaluate, along with a message at the top of the page (which can be changed by an admin) indicating how important evaluating courses is. The course subject code and number for each course will act as a link to the **Course Evaluation** page. These links will be disabled if the evaluation period for the course has not started, if the evaluation period for the course has ended, or if the course has already been evaluated by the user. The user may click the logout button when they are done, though they must exit the browser completely if they want to be completely logged out of SSO.

3.3 Course Evaluation Form

A Student user will be given a set of questions that they may answer voluntarily as well as a section to enter comments. If the student so chooses, they may save the evaluation in order to come back to it later to finish it by clicking the "Save" button. If, however, the user decides to submit the evaluation by clicking the "Submit" button, they will not be able to evaluate that course again, and if the user navigates back to that page after a submission they will be presented with a blank page.

3.4 Admin Home (also known as Home Page)

An Admin user, who has logged in through SSO and has been verified as an admin, will be presented with a home page that gives access to the capabilities of the EWU Online Course Evaluator Web Application. The About section, which each admin page has, will contain information as to what may be done next by an admin in order to reach a particular feature. The tabs at the top of the page, which will be available on all admin pages, allow the admin to navigate to a particular page that has the feature they want to use.

3.5 Admin Management (also known as Admin Page)

An Admin user, who has navigated to the Admin Page, may add or remove admins to/from the listbox that is displayed. However, the admin will not be able to remove themselves from the listbox. Upon clicking the “add” option in the listbox, the user will be presented with a dialog box that will require a name, a username to check against upon a user signing on through SSO, and an email, which is necessary for email notifications. Removal of admins may be achieved by selecting the admin, then clicking the remove button, and then clicking “Accept”. Modifying an admin is done by double-clicking an admin, which will present the user with a dialog box for modifying information.

3.6 Admin Question Management

An Admin user, who has navigated to the required questions page, will be presented with a listbox that displays required questions. By clicking the “Add” button, the user can define a new required question, with a flag turned on or off for retention calculations that is for instructor related questions. Likewise, if the user should choose to modify a question by selecting the question then clicking the “Modify” button, they can simply edit the existing parameters of the question to the their liking. If the question is no longer needed, removal may be achieved by first selecting it and then clicking the “Remove” button. Also, double-clicking a question will result in a modify dialog being displayed.

3.7 Admin Course-Specific Question Management

An Admin user, who has navigated to the Course-Specific Questions page, will be presented with a listbox that displays course-specific questions. The admin by clicking the “Add” button could define a new question that is either of type ‘departmental’, ‘ABET’, or ‘optional’. The user will be able to define the question and then select which courses to apply the question to by checking the appropriate checkboxes. Likewise, if the user should choose to modify a question by selecting the question and then clicking the “Modify” button, they can edit the existing parameters of the question to their liking. If the question is no longer needed, removal may be achieved by first selecting it and then clicking the “Remove” button. Also, double-clicking a question will result in a modify dialog being displayed.

3.8 Admin Instructor Questions Management

An Admin user, who has navigated to the Instructor Questions page, will be presented with a listbox that displays all questions which have been created by instructors. The admin must approve all questions before they are added to an evaluation. The admin can add questions on behalf of the instructors. The admin can then modify any question by selecting it from the listbox and then clicking the “Modify” button. If the question is no longer needed, removal may be achieved by first selecting it and then clicking the “Remove” button. Also, double-clicking a question will also result in a modify dialog being displayed.

3.9 Reports Page

An Admin user, who has navigated to the Reports page, will be presented a listbox of courses for which reports can be generated – all at once or one at time – by double-clicking the desired course. An Admin user can make filter selections to determine which courses will be displayed. Upon the generation of the reports, .pdf files or .csv files, and a .zip file containing the documents, will be downloaded to the Admin user’s local machine. These reports will include: average score of each required question, average of the flagged required questions, average of all required questions, average score of each optional question, average of all optional questions, average of each departmental question, average of all departmental questions, average of each ABET question, average of all ABET questions, and the average of all questions. Each calculation must be from raw data. The report will also include information indicating which course the report is for, what year and quarter it is, the section, and the instructor who is teaching the course. There is another feature of the Reports page by which the instructor of a selected course will be deleted from the database and all the evaluation data for their courses and the question sets for their courses will be as well.

3.10 Instructor’s Question Management

An Instructor user, who has logged in through SSO and has been verified as an Instructor, will be presented with a list of courses to select from in order to manage the questions for the selected course. Upon selecting a course, an Instructor user will be directed to the Question Management page. It is there that the instructor will see listboxes of optional and course-specific questions to include for the course’s evaluation. If they so choose, they may also remove questions from their set of questions for the evaluation. In addition, the instructor may add, modify, or remove course-specific questions that are displayed in the course-specific questions listbox. These questions for the course will be maintained by any instructor who is teaching the course and by the admins, who will be able to edit or remove the questions.

3.11 Information Upload Page

An Admin user, who has navigated to the “Upload” page, will be presented with the status of the current subjects, year, and quarter and the “Prepare the database” button. Upon clicking the button, the courses will be retrieved from the ODATA service and displayed to the admin, who will be able to check the ones that they want to have uploaded to the database. The “Prepare the database” button will also disappear. The start and end dates of the evaluation period for each course is generated upon the retrieval of the course information from the ODATA service. The admin may choose to cancel or submit their selections of the courses retrieved as well as alter the start and end dates of the evaluation period. This data that is uploaded is necessary in order to provide the necessary information to the **Login, Student Home, Admin Question Management, and Instructor Home.**

3.12 Sample Evaluations Page

An Admin user, who has navigated to the “Sample Evaluations” page, will be presented with a list of courses sorted by subject and then by course number. By clicking on one of the course links, the user can view the evaluation form for that course to see what questions are on it.

4. Modification Requirements and Specifications

4.1 Correct Data Retrieval

Specifications:

- Investigate the issue in SQL data retrieval.
- Investigate how data is being displayed.
- Exclude irrelevant courses in a way that's still flexible.
- Have the quarter indicator properly update.

4.2 Correct Course Evaluation Opening/Closing

Specifications:

- Investigate where the automated opening/closing system is located and how it works.
- Add manual control to automated system for opening and closing.
- Stop a student submission from closing the entire system and instead just close the class they are evaluating.

4.3 Increase Accessibility of Questions for Admins

Specifications:

- Design a feature that allows the admin to add questions for instructors.
- Create and implement the feature.

4.4 Increase Accessibility of Notices

Specifications:

- Investigate where hardcoded notices are located.
- Design a system to allow for modification of notices by an Admin.
- Create and implement the system.

4.5 Redesign Web UX (User Experience)

Specifications:

- Investigate what needs to be altered in UX design.
- Define layouts per page.
- Define color scheme.
- Redesign Student evaluation page.

- Redesign Admin home page.

4.6 Implement Admin Review of Faculty Questions

Specifications:

- Investigate submission of faculty questions.
- Design a system to intercept faculty questions and submit them to the Admin, and have the Admin confirm them for publishing.
- Create and implement the system.

4.7 Implement Unit Testing Suite

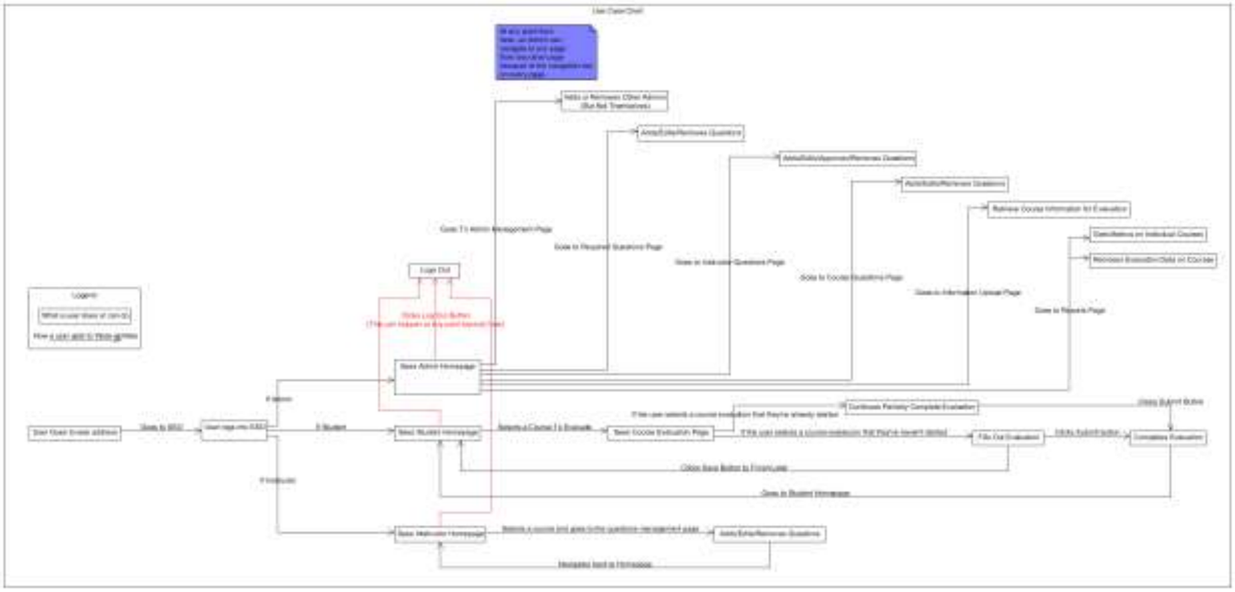
Specifications:

- Investigate what methods are available for unit testing in PHP.
- Create and implement unit testing methods.

5. Document Changes and Reflections

Our current SRS document has not changed much when compared to the previous versions. It was mainly built off of the existing SRS from the previous team. After meeting with Margo, we added the new requirements for the project and cleaned up what grammatical errors we could find. For our final version, we cleaned up the grammatical errors that we missed and unified the layout and fonts in the document. Lastly, we removed the “Future Enhancements” section of the document, because we do not wish to be held accountable for any of the optional features if they are not implemented.

6. Use Case Chart


Use Case Chart
Adds or Removes Other Admins (But Not Themselves)
Adds/Edits/Removes Questions
Adds/Edits/Approves/Removes Questions
Adds/Edits/Removes Questions
Retrieve Course Information for Evaluation
Get Statistics on Individual Courses
Retrieves Evaluation Data on Courses
Goes To Admin Management Page
Goes to Required Questions Page
Goes to Instructor Questions Page
Goes to Course Questions Page
Goes to Information Upload Page
Goes to Reports Page
Log Out
Clicks Logout Button
Legend
What a user does or can do
How a user gets to those actions
Sees Admin Homepage
If Admin
Goes to SSO
User logs into SSO
If Student
Sees Student Homepage
Selects a Course To Evaluate
Sees Course Evaluation Page
If the user selects a course evaluation that they're already started
Continues Partially Complete Evaluation
Clicks Submit Button
If the user selects a course evaluation that they haven't started
Fills Out Evaluation
Clicks Submit button
Completes Evaluation
Clicks Save Button to Finish Later
Goes to Student Homepage
If Instructor
Sees Instructor Homepage
Selects a course and goes to the questions management page
Adds/Edits/Removes Questions
Navigates back to Homepage